Microsoft

.net developer training

Create XML Web Services and Applications with Visual Studio .NET

Lifecycle Tools for .NET

Module 12 Agenda

- Lifecycle Tools for .NET
  - Overview
  - Enterprise Templates
  - Software Modeling Using UML
  - Database Modeling Using ORM
  - Application Center Test
  - Visual SourceSafe

Lifecycle Tools for .NET
Overview

- Enterprise development is systematic
- Collaborative work environment
- Support for all roles essential
- Standards help – e.g., UML, XML, etc.
- Architectural guidance and reusable assets essential

Lifecycle Tools for .NET

- Enterprise Templates
- Software Modeling Using UML
- Database Modeling Using ORM
- Application Center Test
- Visual SourceSafe

Lifecycle Tools for .NET

- **Enterprise Templates**
- Software Modeling Using UML
- Database Modeling Using ORM
- Application Center Test
- Visual SourceSafe™

Enterprise Templates
Goals

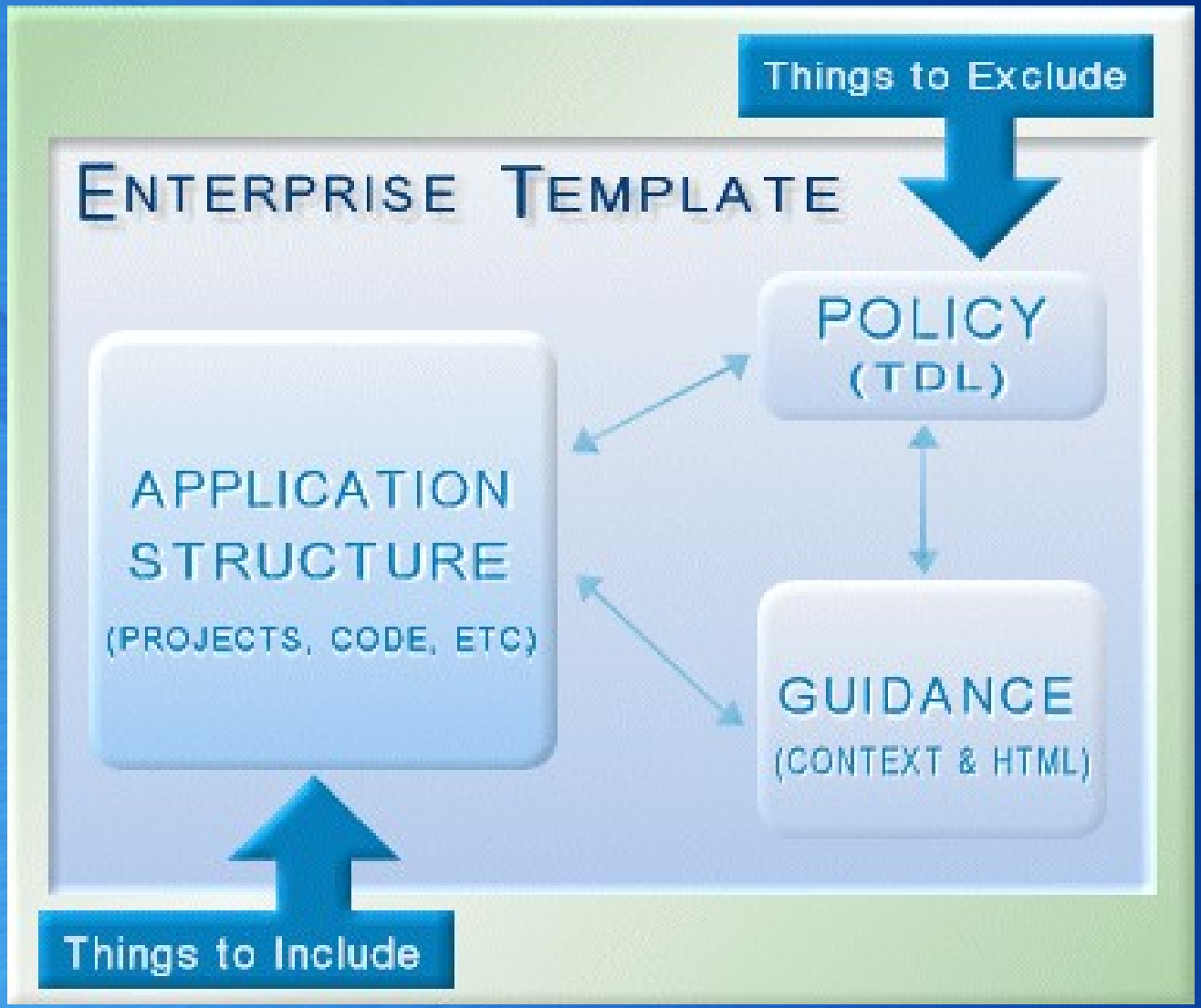
Share corporate knowledge & experience at design time to:

- **Reduce the “Blank Slate” problem**
- **Reduce the “Full Slate” problem**
- **Make it easier to do the right thing**

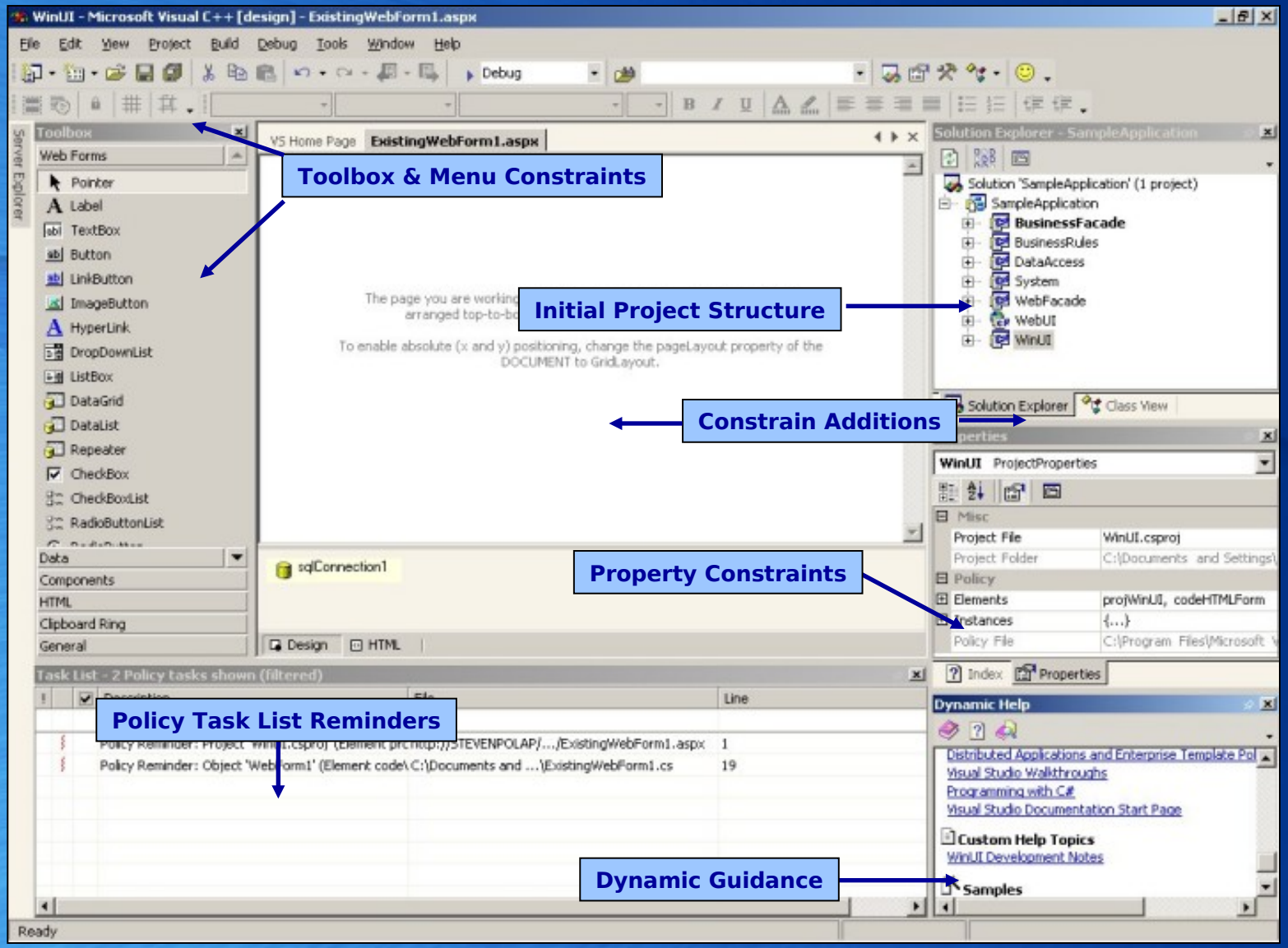
Enterprise Templates
Effective Architectural Guidance

- **Enterprise Templates**
  - Solve “blank state” and “full state” problem
  - Provides infrastructure and content for best practices
- **Enterprise Template Structure**
  - Initial Project Structure or Template Project
    - Establish blue-prints for an application
  - Policy Definition file
    - Solve “full state” problem

Enterprise Template
Components of Enterprise Template



Enterprise Template
Visual Studio.NET IDE Features



Enterprise Templates
Summary

- Organize structure for complex projects
- Supply pre-built common implementation
- Simplify IDE for newer developers
- Help avoid common mistakes
- Predefine common property settings
- “Just In Time” focused guidance

Lifecycle Tools for .NET

- Enterprise Templates
- Software Modeling Using UML
- Database Modeling Using ORM
- Application Center Test
- Visual SourceSafe

Software Modeling
Features Overview

- Unified Modeling Language (UML) Support
- Error Checking
- Reporting and Analysis

Software Modeling Using UML

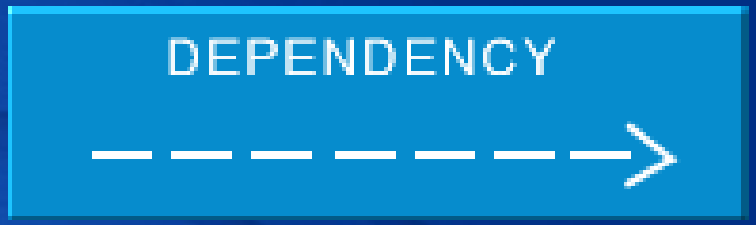
- Software projects produce many artifacts
  - Requirements
  - Architecture
  - Design
  - Source Code etc…
- UML
  - Language for constructing, visualizing, and documenting artifacts
  - Elements of the architecture and details
  - Relationships between elements
  - Diagrams of elements and their relationships

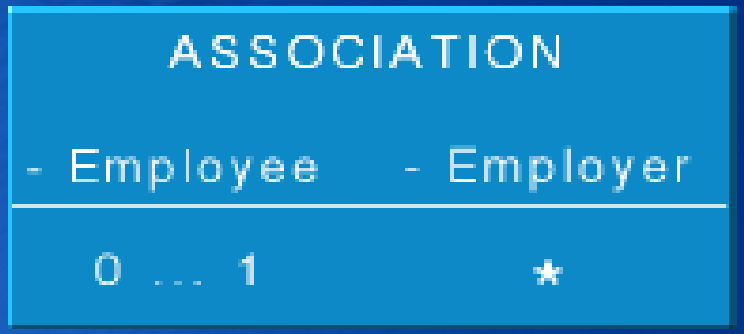
Software Modeling Using UML
Building Blocks

- **Elements**
  - Basic building blocks of a model
  - Types
    - Structural Elements
    - Behavioral Elements
    - Grouping Elements
    - Annotational Elements
- **Relationships**
  - Associate various elements
- **Diagrams**
  - Visually organizes collections of elements and relationships

Software Modeling Using UML
Relationships

- **Dependency**
  - Indicates a change in specification of one changes another, but not the reverse
- **Generalization**
  - Indicates a relationship as superclass/subclass or parent class/child class
- **Association**
  - Specifies objects of one entity are connected to another





Software Modeling Using UML
Diagram Types

- **Structural Diagrams (Static)**
  - Class Diagram (static structure)
  - Component diagram
  - Deployment diagram
- **Behavioral Diagrams (Dynamic)**
  - Use Case diagram
  - Sequence diagram
  - Collaboration diagram
  - State Chart diagram
  - Activity diagram

Software Modeling
Code Generation and Reverse Engineering

- Generate code for specific classes
- Generate code as part of a new project
  - Windows Forms, ASP.NET, Class Library, Web Service, etc.
- Generate code in a specific language:
  - C#
  - C++
  - Visual Basic.NET
- Generate code in a specific language:
  - Generate code based on code templates
  - Reverse Engineering

Software Modeling Demo

- Create a UML Class in Visio
- Generate Code from a UML Class in VB.NET
- Error Checking of UML Model

Lifecycle Tools for .NET

- Enterprise Templates
- Software Modeling Using UML
- **Database Modeling Using ORM**
- Application Center Test
- Visual SourceSafe™

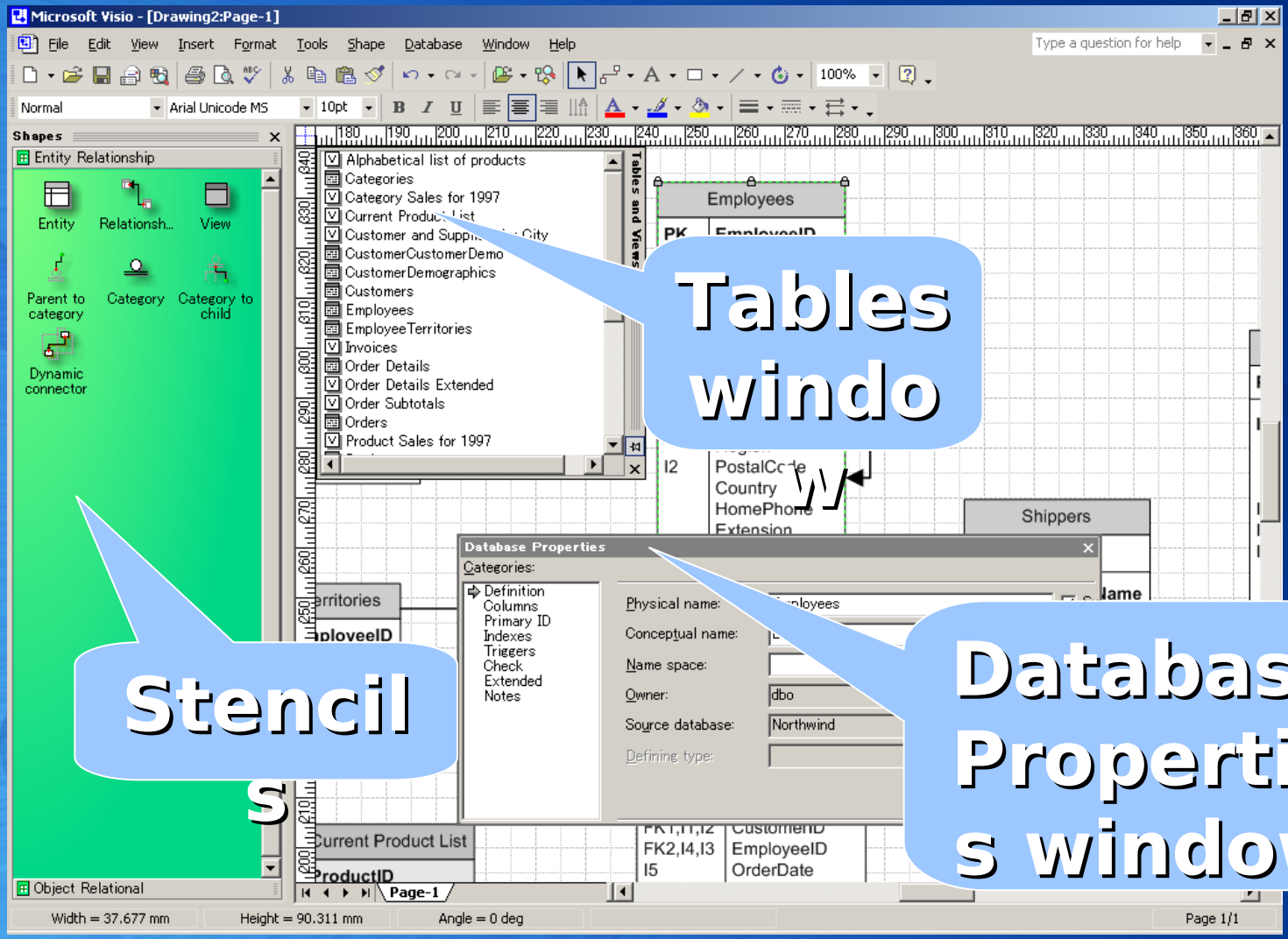
Database Modeling
Features Overview

- Visio Integrated with Visual Studio.NET IDE
- Entity relationship (ER) diagram creation using multiple notations
- Database schema generation, either directly or through a DDL script
- Database schema reverse engineering from leading RDBMS
- Model error checking
- Model-database synchronization
- Database reports
- Workgroup collaboration
- Business rules-based database design (ORM)

Database Modeling
Object Role Modeling (ORM)

- Conceptual database design using natural language facts and examples
- Logical and physical database models automatically built based on natural language facts
- Rules Expression
- Advantages of ORM:
  - Easy to describe relationship between objects
  - Easily adapts to changes in system
  - Easy to query a database

Database Modeling
Environment – ER Diagram



Database Modeling
Reverse Engineering

- Use the Reverse Engineering Wizard to:
  - Generate ER Models
  - Generate ORM Models
  - Extract tables, views and stored procedures
  - Populate the tables window
  - Extract code and add it to the model
  - Document existing database structures
- Drag tables (or views) onto diagram page
  - Relationships are drawn automatically
- Modify code and attributes of the

Database Modeling Demo

- Generate a Logical Model from an ORM Schema
- Generate database from the ER Diagram

Lifecycle Tools for .NET

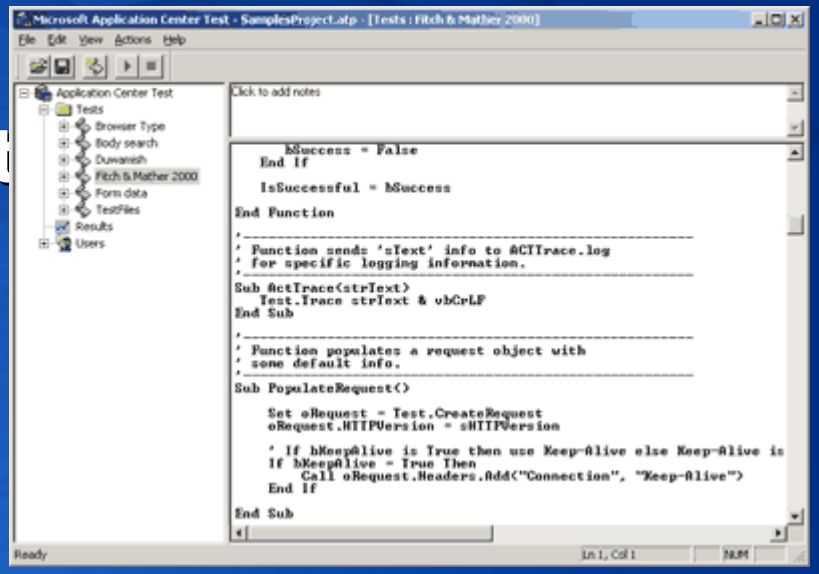
- Enterprise Templates
- Software Modeling Using UML
- Database Modeling Using ORM
- **Application Center Test**
- Visual SourceSafe

Application Center Test
Features

- Integrated with Visual Studio .NET IDE
- Test Web Services and Applications
  - Stress & functional testing
- Easy to create and run tests
- Open Extensible Architecture
- Rich Customizable Reporting
  - Built on XML

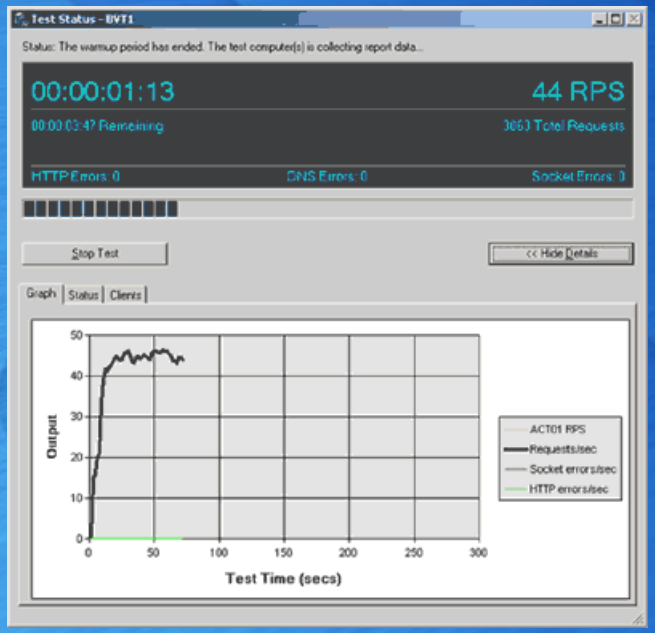
Application Center Test
Test Configuration

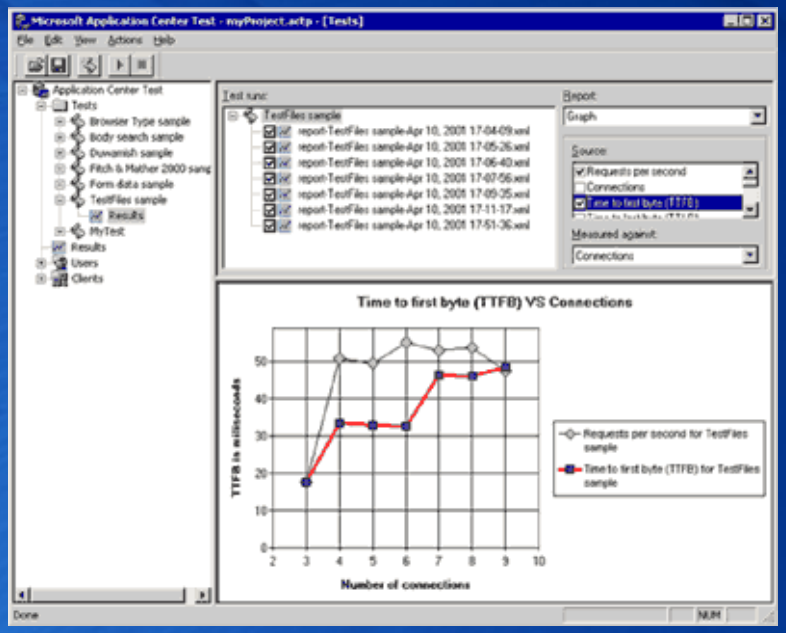
- Creating Tests
  - Browser Record
  - Importing a Web Server Log File
- Configuring Tests
- Running Tests
- Analyzing the Result
- Automating Tests



Application Center Test Demo

- Use Application Center Test to create and run tests
- Analyze the Application Center Test results





Lifecycle Tools for .NET

- Enterprise Templates
- Software Modeling Using UML
- Database Modeling Using ORM
- Application Center Test
- **Visual SourceSafe**

Visual SourceSafe
Features

- **Version Control**
  - Versioning and Configuration Management
  - Source Control System
- **Tracking Changes**
  - Pending Check-ins Window
- **Parallel Development**
  - Multiple development environments

Lifecycle Tools for .NET - Summary

- Enterprise Templates for Architectural Guidance
- UML Software Modeling
- ORM Data Modeling
- Application Center Test
- Visual SourceSafe

Resources

- http://msdn.microsoft.com/vstudio/nextgen/Technology/entfeatures.asp
- http://msdn.microsoft.com/vstudio/techinfo/articles/
- Object Role Modeling resource

  www.orm.net
- Application Center Test resource

  www.microsoft.com/applicationcenter

12.34